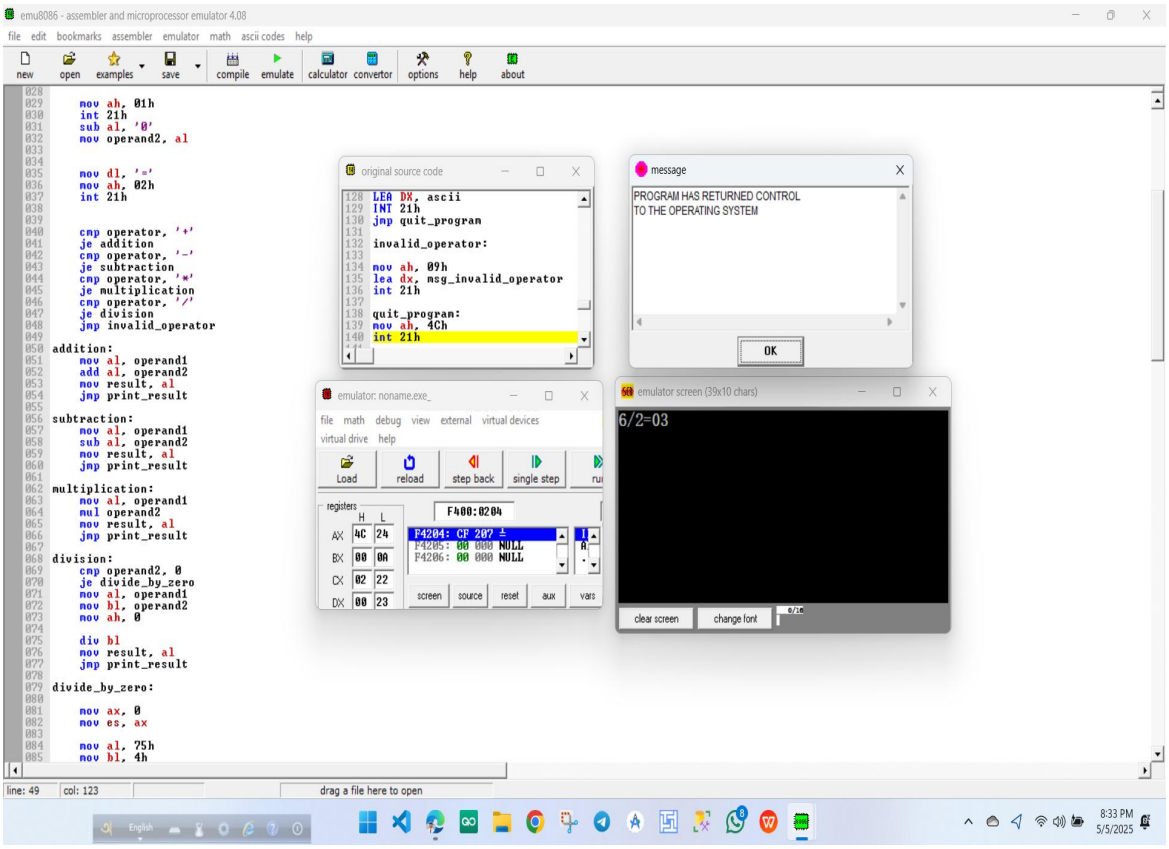
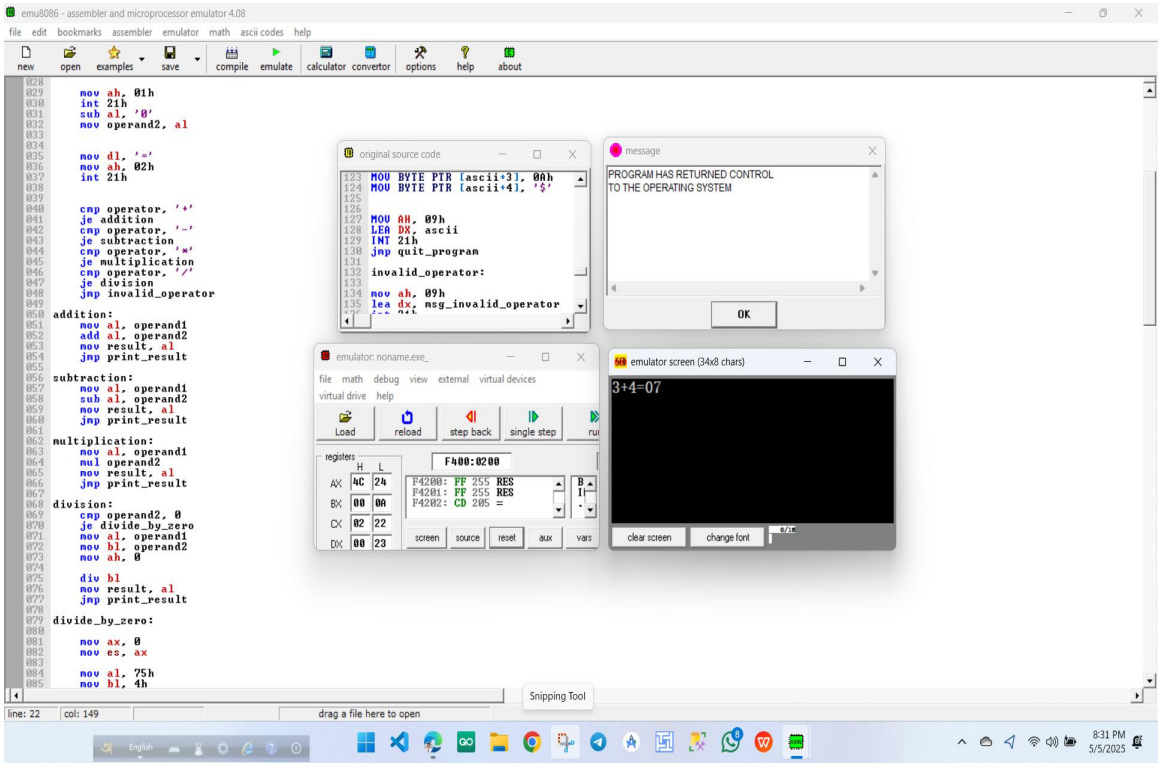
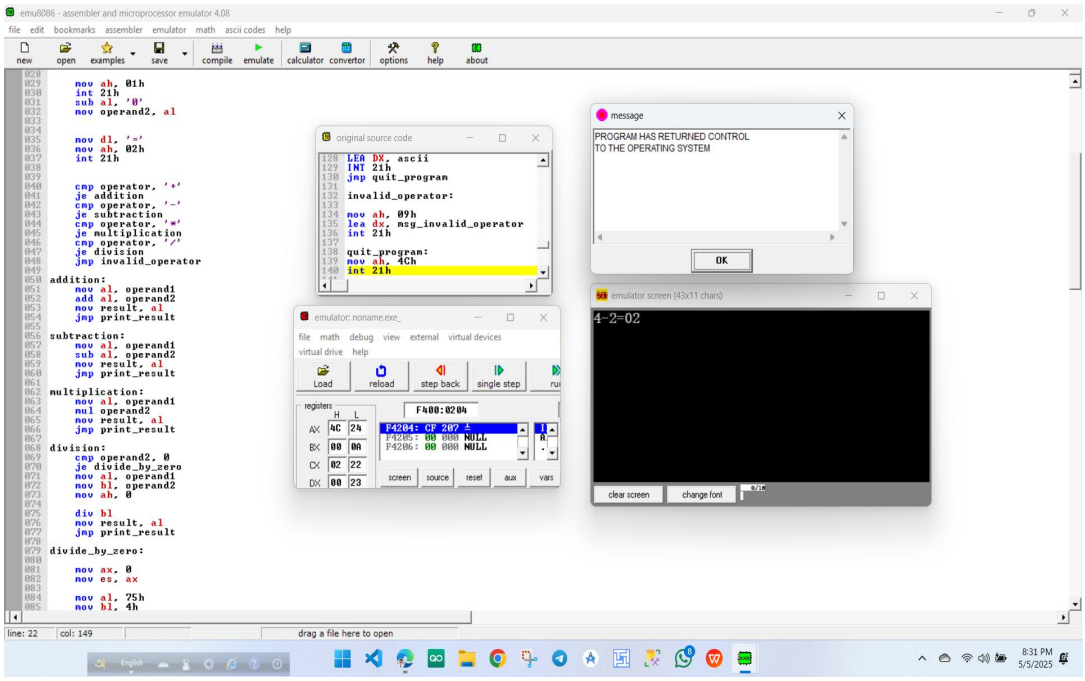
```
.stack 100h

.data
operand1 db ?
operand2 db ?
result db ?
operator db ?
msg_divide_by_zero db "Infinity", 0Dh, 0Ah, "$"
msg_invalid_operator db "Invalid operator", 0Dh, 0Ah, "$"
negative_sign db "-"
ascii db 2 DUP(?)

.code
main proc
   mov ax, @data
   mov ds, ax


   mov ah, 01h
   int 21h
   sub al, '0'
   mov operand1, al


   mov ah, 01h
   int 21h
   mov operator, al

   mov ah, 01h
   int 21h
   sub al, '0'
   mov operand2, al


   mov dl, '='
   mov ah, 02h
   int 21h


   cmp operator, '+'
   je addition
   cmp operator, '-'
   je subtraction
   cmp operator, '*'
   je multiplication
   cmp operator, '/'
   je division
   jmp invalid_operator

addition:
   mov al, operand1
   add al, operand2
   mov result, al
   jmp print_result

subtraction:
   mov al, operand1
```

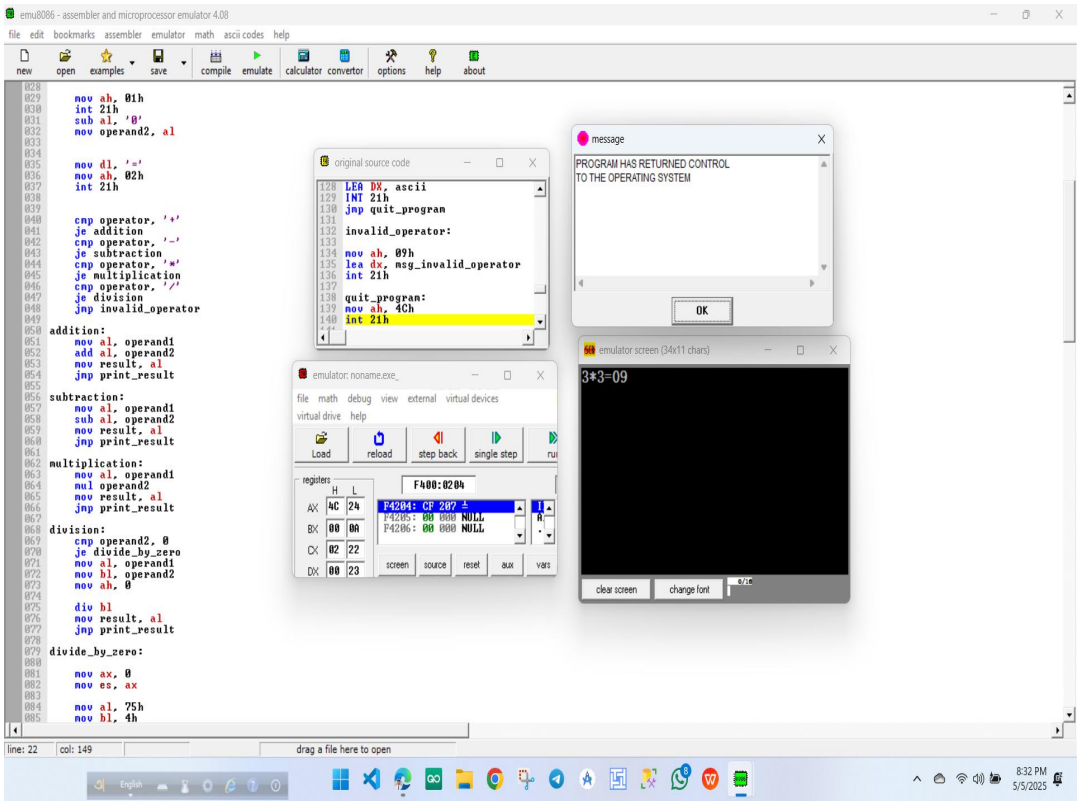
```
    sub al, operand2
    mov result, al
    jmp print_result

multiplication:
    mov al, operand1
    mul operand2
    mov result, al
    jmp print_result

division:
    cmp operand2, 0
    je divide_by_zero
    mov al, operand1
    mov bl, operand2
    mov ah, 0

    div bl
    mov result, al
    jmp print_result

divide_by_zero:

    mov ax, 0
    mov es, ax

    mov al, 75h
    mov bl, 4h
    mul bl
    mov bx, ax

    mov si, offset [infinity_msg]
    mov es:[bx], si
    add bx, 2
    mov ax, cs
    mov es:[bx], ax

    int 75h
    jmp quit_program

print_result:

    cmp result, 0
    jns print_result_positive

    mov dl, negative_sign
    mov ah, 02h
    int 21h

    neg result

print_result_positive:
    MOV AL, result
    MOV AH, 0
    MOV BH, 0
    MOV BL, 10
```

```
    DIV BL
    ADD AL, '0'
    MOV ascii[0], AL
    ADD AH, '0'
    MOV ascii[1], AH


    MOV BYTE PTR [ascii+2], 0Dh
    MOV BYTE PTR [ascii+3], 0Ah
    MOV BYTE PTR [ascii+4], '$'


    MOV AH, 09h
    LEA DX, ascii
    INT 21h
    jmp quit_program

invalid_operator:

    mov ah, 09h
    lea dx, msg_invalid_operator
    int 21h

quit_program:
    mov ah, 4Ch
    int 21h

main endp


infinity_msg PROC

    mov ah, 09h
    lea dx, msg_divide_by_zero
    int 21h
    IRET

infinity_msg ENDP


end main
```


emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source code
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
OK
emulator: noname.exe_
emulator screen (39x10 chars)
6/2=03
Load reload step back single step
clear screen change font
line: 49 col: 123
drag a file here to open


emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source code
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
OK
emulator: noname.exe_
emulator screen (34x8 chars)
3+4=07
Load reload step back single step
clear screen change font
line: 22 col: 149
drag a file here to open
Snipping Tool


emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source code
LEA DX, ascii
INT 21h
jmp quit_program
invalid_operator:
mov ah, 09h
lea dx, msg_invalid_operator
int 21h
quit_program:
mov ah, 4Ch
int 21h
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
OK
emulator: noname.exe_
file math debug view external virtual devices
virtual drive help
Load reload step back single step
registers
F400:0204
F4204: CF 207
F4205: 00 000 NULL
F4206: 00 000 NULL
screen source reset aux vars
emulator screen (43x11 chars)
4-2=02
clear screen change font
line: 22 col: 149
drag a file here to open
8:31 PM
5/5/2025


emu8086 - assembler and microprocessor emulator 4.08
file edit bookmarks assembler emulator math ascii codes help
new open examples save compile emulate calculator convertor options help about
original source code
LEA DX, ascii
INT 21h
jmp quit_program
invalid_operator:
mov ah, 09h
lea dx, msg_invalid_operator
int 21h
quit_program:
mov ah, 4Ch
int 21h
message
PROGRAM HAS RETURNED CONTROL
TO THE OPERATING SYSTEM
OK
emulator: noname.exe_
file math debug view external virtual devices
virtual drive help
Load reload step back single step
registers
F400:0204
F4204: CF 207
F4205: 00 000 NULL
F4206: 00 000 NULL
screen source reset aux vars
emulator screen (34x11 chars)
3*3=09
clear screen change font
line: 22 col: 149
drag a file here to open
8:32 PM
5/5/2025